MAPAS AUTO-ORGANIZÁVEIS EM VEÍCULOS AUTÔNOMOS SUBAQUÁTICOS

Silvia Silva da Costa Botelho[1], Celina Häffele da Rocha[2], Mônica da Silva Figueiredo[3], Gabriel Leivas Oliveira[4], Paulo Lilles Drews Junior[5]

[1] Professora Doutora em Informática e Telecomunicações (FURG) – silviacb@furg.br

[2] Mestranda em Modelagem Computacional (FURG) – celinahaffele@yahoo.com.br

[3] Acadêmica do Curso de Engenharia de Computação (FURG) – monica.sfigueiredo@hotmail.com

[4] Mestrando em Ciências da Computação (UFMG) – gabrielleivas@gmail.com

[5] Mestre em Ciência da Computação (UFMG) – dudopel@gmail.com

Resumo

O uso de veículos autônomos subaquáticos (AUVs) para tarefas submarinas é um campo promissor da robótica. Estes robôs podem transportar uma câmera de inspeção visual, que além de inspecionar e mapear, as imagens capturadas podem auxiliar a navegação e localização dos robôs. Neste contexto, este trabalho propõe uma abordagem para o mapeamento destes veículos. Supondo o uso de câmeras de inspeção, esta proposta é composta pelo desenvolvimento de mapas topológicos utilizando mapas auto-organizáveis e estruturas celulares crescente (GCS) para a localização e navegação. Uma série de testes foram realizados, em relação a problemas de desempenho online. Os resultados revelaram uma boa precisão e robustez para uma série de condições subaquáticas, como iluminação e ruído, mostrando ser uma técnica de mapeamento visual promissora e original.

Palavras-chaves – mapas auto-organizáveis; estruturas celulares crescentes; mapeamento; navegação, localização.

SELF ORGANIZING MAPS FOR AUVS MAPPING

Abstract

The use of Autonomous Underwater Vehicles (AUVs) for underwater tasks is a promising robotic field. These robots can carry visual inspection cameras. Besides serving the activities of inspection and mapping, the captured images can also be used to aid navigation and localization of the robots. In this context, this paper proposes an approach to mapping of underwater vehicles. Supposing the use of inspection cameras, this proposal is composed of the development of topological maps using self-organizing maps and Growing Cell Structures (GCS) for localization and navigation. A set of tests was

accomplished, regarding online and performance issues. The results reveals an accuracy and robust approach to several underwater conditions, as illumination and noise, leading to a promissory and original visual mapping technique.

Keywords – self-organizing maps; growing cell structures; mapping; navigation, localization.

1 – INTRODUCTION

Autonomous Underwater Vehicles (AUVs) are mobile robots that can be applied to many tasks of difficult human exploration [1]. In underwater visual inspection, the vehicles can be equipped with down-looking cameras, usually attached to the robot structure [2]. These cameras capture images from the deep of the ocean. In these images, natural landmarks, also called key points in this work, can be detected allowing the AUV localization and mapping.

In this paper we propose a new approach to AUV mapping. Our approach extract and map key points between consecutive images in underwater environment, building online key points maps. This maps can be used to robot localization and navigation. We use Scale Invariant Feature Transform (SIFT), which is a robust invariant method to key points detection [3]. Furthermore, these key points are used as landmarks in an online topological mapping.

We propose the use of self-organizing maps (SOM) based on Kohonen maps [4] and Growing Cell Structures (GCS) [5] that allow a consistent map construction even in presence of noisy information.

First the paper presents a detailed view of the SOM and GCS. Next, our approach is presented, followed by the implementation, test analysis and results with different undersea features. Finally, the conclusion of the study and future perspectives are presented.

2 – USING SOM AND GCS FOR AUV MAPPING

Figure 1 shows an overview of the approach proposed here. First, the underwater image is captured and pre-processed to removal distortions caused by water diffraction [6]. With the corrected image, key points are detected and local descriptors for each one of these points are computed by SIFT. Each key point has a n dimensional local descriptors and global pose informations. A matching stage provides a set of correlated key points between consecutive images. The relative motion between frames is estimated, using the correlated points and the homography matrix [7].

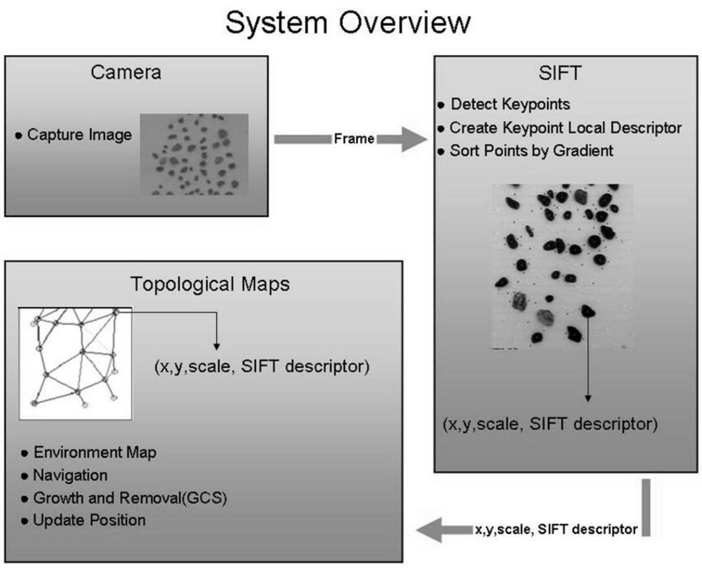

FIGURE 1: System Overview

The key points are used to create and train the topological maps. The growing cell structures algorithm is used to create the nodes and edges of the SOM. Each node has a n-dimensional weight. After a training stage, the system provides a topological map, where its nodes represent the main key points of the environment.

During the navigation, when a new image is captured, the system calculus its local descriptors, correlating them with the nodes of the current trained SOM.

Next, it is detailed the proposed approach.

2.1 – Self-Organized Maps

Self-Organizing maps are neural grids based on competition and unsupervised learning. The process starts from a competition layer which is usually uni or two-dimensional [8]. Each neuron of the respective layer is connected with a input layer through weights. If the input layer has X neurons, each neuron of competition layer will have X connections with the neurons of input layer, see Figure 2.

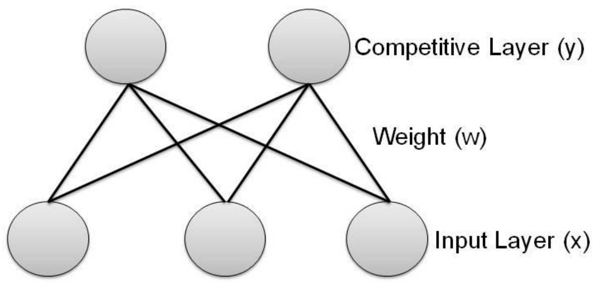

FIGURE 2: Network Model

To obtain the winner neuron, which better represents input data, neurons of competitive layer fight among themselves. The neurons of competitive layer compete to be the neuron that best represents the input data, called the winner neuron. For this it is used a similarity metric, for instance, Euclidean Distance between the input vector and the vector of synaptic weights of the neuron in question, which derived from the equation 1.

$$d_i(t) = \sum_{j=1}^{n} (x_j(t) - w_{ij}(t))^2 \quad \textbf{(1)}$$

where x = (x_1, x_2,…, x_n), represents a set of input data in a time t and w = (w_1, w_2,…, w_n) represents the set of weights of neuron *i* in time *t*. The organization of neurons with their neighbors are called neighborhood. The best self-organization is obtained when the set of neighbors get extensive and decreases monotonically with time. A learning function is a multiplying of a learning constant ∝ by the difference between input vector (X) and weights (W), plus the weight vector (W) applied the winner neuron and all who were in your neighborhood, mathematically, is obtained from the equations 2 and 3.

$$W_i(t+1) = W_i(t) + \alpha(t) * [X(t)] \text{, if } i \in V_i(t) \quad \textbf{(2)}$$

$$W_i(t+1) = W_i(t) \text{, if } i \notin V_i(t) \quad \textbf{(3)}$$

SOM - Basic Algorithm

The algorithm for the formation of self-Organizing maps follows the process of *competition, cooperation and synaptic adaptation*. First, the competition, for each input pattern, calculates the response of output neurons (grid).

The neuron with the highest response is the winner of the competition (Euclidean distance). In cooperation the winner neuron defines a topological neighborhood of excited neurons. Finally, the synaptic adaptation is learning from the standard input. The weights of the winner neuron and its neighborhood are closer to the standard input. The following will be presented more detailed descriptions of the processes.

The algorithm starts with a high level of neighborhood and this is reduced as time progresses, it is necessary to reduce the region's neighborhood for the self-organizing map, Figure 3. Gaussian function is a interesting way to implement a neighborhood function, once that is invariant to translation.

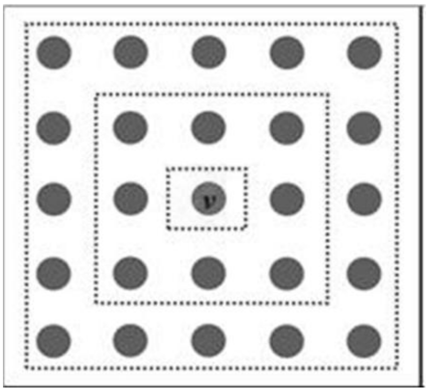

FIGURE 3: Neighborhood Region

In the adaptation is performed a modification on weights in relation to input, iteratively. The equation is applied to whole grid of neurons inside neighborhood region $h_{j\,i}$, as noted in equation 4.

$$w_j(t+1) = w_j(t) + \alpha(t)h_{jii}(x)(t)(x - w_j(t)) \qquad \textbf{(4)}$$

where $w_j(n + 1)$ is updated weight, $w_j(n)$ the previous weight vector, α learning rate $h_{ji}(x)$ neighborhood and x $w_j(t)$ is adaptation.

2.2 – GCS - Growing Cell Structures

A cell structure is a self-organizing map with one important feature, which is the ability to find by yourself a specific network for the problem through a growth process. In the proposed approach, we use Growing Cell Structures (GCS) proposed by Bernd Fritzke [5], searching for solve the greatest restriction of Kohonen nets, a fix topology, whereas to define a optimal size of network it's necessary obtain some a priori knowledge, that is generally unavailable [9]. Without these knowledge you can limit the ability of the network.

This type of SOM with variable topology allows the network to grow from an initial topology minimal n-dimensional. Regardless of the size of the initial topology, each cell containing a vector of the size of mapped space. These structures are able to grow against the need for better classification of points in space. Each number of misclassifications, a new cell is created. This incorrect classification is defined as the distance vector presented with the best vector found in the map, where more than a specified value is set to misclassification.

2.2.1 – Cell Insertion: If a number of input signals exceeds a threshold value, a new cell W_{new} is inserted between the cell that has the greatest number of winning times W_{bmu} and its nearest neighbor W_{prox}, Figure 4, where the node in gray is the cell to be inserted. The weight of the new unit is given by equation 5.

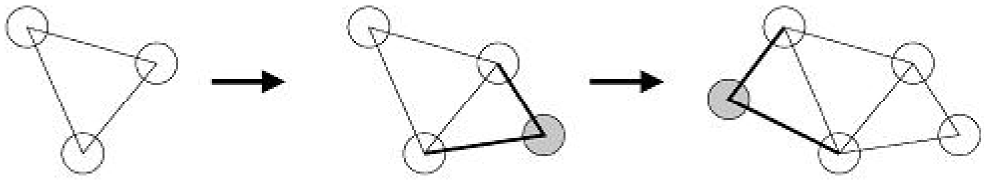

FIGURE 4: Illustration of cell insertion

$$W_{new} = (W_{bmu} + W_{prox})/2 \qquad \textbf{(5)}$$

2.2.2 – Cell Deletion: To remove nodes, after a certain number of iterations, the cell with the highest average Euclidean distance will be removed and all its neighbors that are left with only one connection after its removal will also be excluded, see Figure 5.

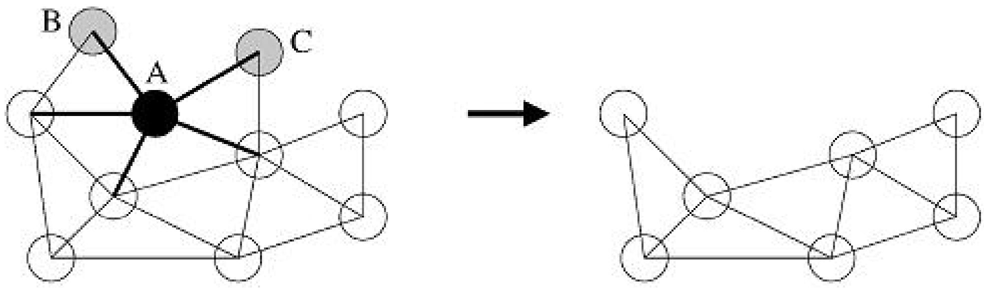

FIGURE 5: Illustration of cell deletion: Cell A is deleted. Cells B and C are within the neighborhood of A and would be left dangling by removal of the five connections surrounding A, so B and C are also deleted

Integrating the SOM concepts and the GCS proposal, our approach allows a dynamical map building. After a training set, the topological map represents the space of descriptors. The nodes converge to the ideal set of percentage informations. This map can be update during the navigation, enhancing the accuracy of the environment representation.

3 – IMPLEMENTATION, TESTS AND RESULTS

In this work we use the robot presented in Figure 6. This robot is equipped with a Tritech Typhoon Colour Underwater Video Camera with zoom, a miniking sonar and a set of sensors (altimeters and accelerometers) [10].

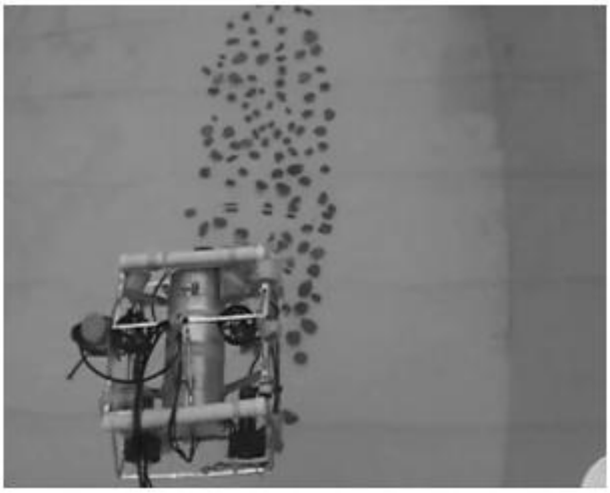

FIGURE 6: ROVFURGII in test field

The visual system was tested in a desktop Intel Core 2 Quad Q6600 computer with 2Gb of DDR2-667 RAM. The camera is NTSC standard using 320x240 pixels at a maximum rate of 29.97 frames per second.

A group of tests was conducted with videos captured by ROV. The table I shows the steps of an intermediate map adaptation with its number of frames, points of interest (captured by the visual sensor) [11] and nodes. The illustration 7 illustrates the growth of the map with 50 frames presented by the visual sensor, below it illustrates the state of the map with 120 frames and the last with 170 frames.

TABLE I

Map construction with GCS algorithm

Frames	Key points	Nodes
50	5108	76
120	9360	198
171	11628	287
342	23256	386
684	46512	545
1026	69768	683

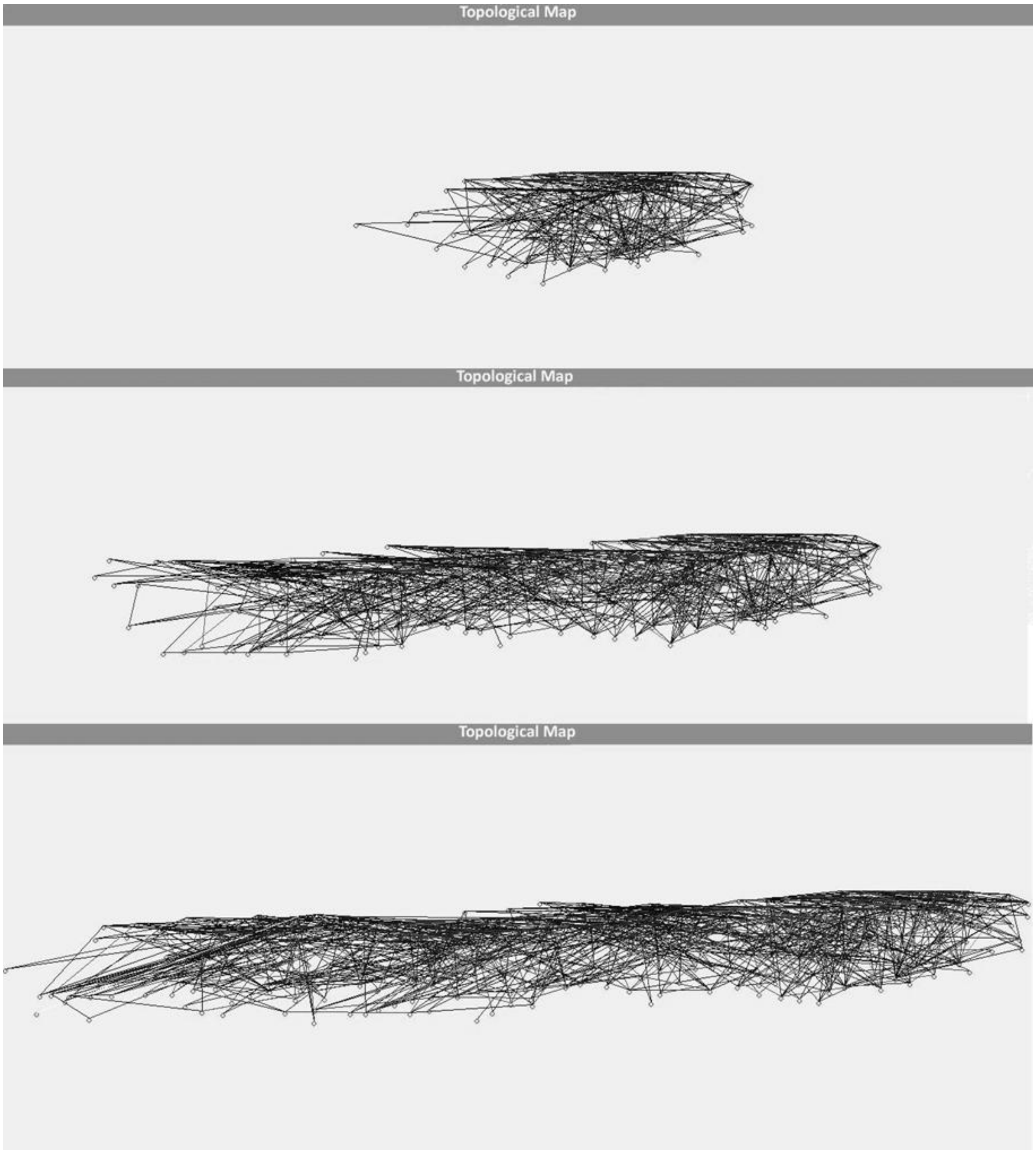

FIGURE 7: State of the topological map in three different states.

The Figure 8 shows the map after the test phase (1026 frames), we can notice that there is not great modification starting from the frame 170, because from this point, it happens a replay of the points already contained in the map, but with accumulated error. Our approach obtained as results a reduction of the error in the localization when we use the topological approach in relation to the use of simple visual odometry.

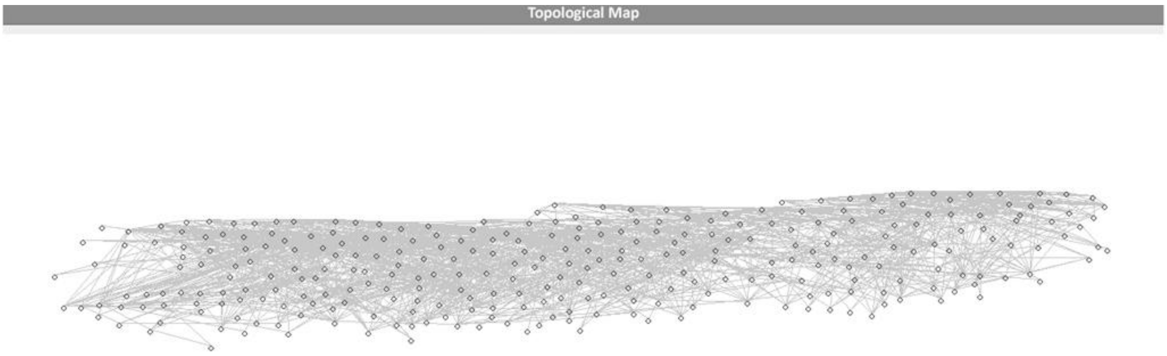

FIGURE 8: State of the final topological map.

We can verify this reality in the illustration 9. Which shows the robot's position in a navigation task. In this task, it passed three times on the reference. In this illustration are represented the estimated position of both approaches, in blue the topological and in red the visual odometry. The table II shows the normalized mistake of position of each method.

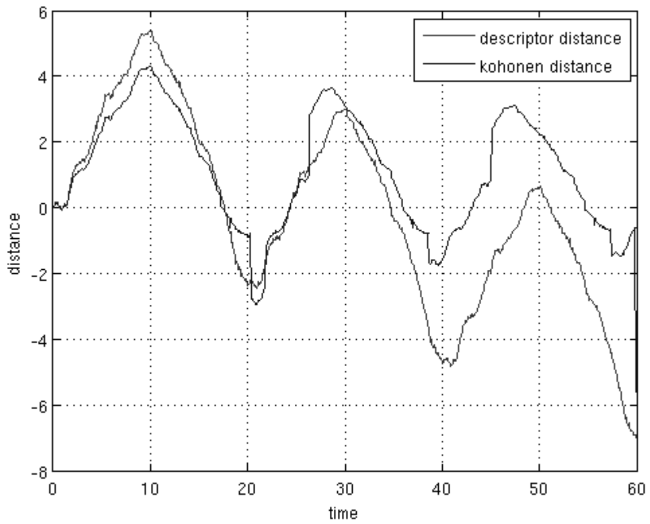

FIGURE 9: Distance Y generated by ROVFURGII in movement.

TABLE II

Normalized error of localization of visual odometry e SOM

Visual Odometry	SOM
0.33	0.09
0.68	0.35
1.00	0.17

The reduction of error associated to the localization with the SOM validates the robustness of the topological approach. Then the topological map in its final state allows us to navigate in two different ways: Through positions as objective and for visual goals. Starting from the current position, search algorithms in graphs such as Dijkstra [12] or the algorithm A* [13] can be used to find a path to the target place.

4. CONCLUSION

This paper details a new approach for mapping of a underwater robot using visual information, SOM and GCS. This system can be used either in autonomous inspection tasks or in control assistance of robot closed-loop, in case of a human remote operator.

The effectiveness of our proposal was evaluated inside a set of real scenarios. The original integration of visual information (using SIFT descriptors) and topological maps with GCS for AUV navigation is a promising field. The topological mapping based on Kohonen Nets and GCS showed potential to underwater SLAM (Simultaneous Localization and Mapping) applications using visual information due to its robustness to sensory impreciseness and low computational cost. The GCS stabilizes in a limited number of nodes sufficient to represent a large number of descriptors in a long sequence of frames. The SOM localization shows good results, validating its use with visual odometry.

As future work, we continue to detail the analysis of our topological mapping system, executing a set of tests with different scenarios and parameters. We intend to fusion different sensor information.

REFERENCES

[1] S.D. Fleischer, “Bounded-error vision-based navigation of autonomous underwater vehicles,” Ph.D. dissertation, Stanford University, 2000.

[2] R. Garcia, V. Lla, and F. Charot, “Vlsi architecture for an underwater robot vision system,” in IEEE Oceans Conference, vol. 1, 2005, pp.674–679.

[3] D. Lowe, “Distinctive image features from scale-invariant key points,” International Journal of Computer Vision, vol. 60, no. 2, pp. 91–110,2004.

[4] T. Kohonen, Self-Organizing Maps. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2001.

[5] B. Fritzke, “Growing cell structures - a self-organizing network for unsupervised and supervised learning,” TR-93-026, University of California- Berkeley, International Computer Science Institute, Tech. Rep., May1993.

[6] N. R. Gracias, “Towards detecting changes in underwater image sequences,” MTS/IEEE Oceans08 Conference, pp. 460–471, 2008.

[7] P. H. S. Torr and D. W. Murray, “The development and comparison of robust methods for estimating the fundamental matrix,” International Journal of Computer Vision, vol. 24, no. 3, pp. 271–300, 1997.

[8] J.A. Freeman and D.M. Skapura, Neural Networks: Algorithms, Applications, and Programming Techniques. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 1991.

[9] B. Fritzke, “Kohonen feature maps and growing cell structures - a performance comparison,” in Advances in Neural Information Processing Systems 5. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993, pp. 123–130.

[10] M. Centeno, “Rovfurg-ii: Projeto e construção de um veículo subaquático não tripulado de baixo custo,” Master’s thesis, Engenharia Oceânica - FURG, 2007.

[11] O. Booij, B. Terwijn, Z. Zivkovic, and B. Krose, “Navigation using an appearance based topological map,” in IEEE International Conference on Robotics and Automation, April 2007, pp. 3927–3932.

[12] E. W. Dijkstra, “A note on two problems in connexion with graphs, ”Numerische Mathematik, vol. 1, pp. 269–271, 1959.

[13] R. Dechter and J. Pearl, “Generalized best-first search strategies and the optimality af a*,” Journal of the Association for Computing Machinery, vol. 32, no. 3, pp. 505–536, July 1985.